

**AMENDMENTS TO THE CLAIMS:**

This listing of claims will replace all prior versions, and listings, of claims in the application:

**Listing of Claims:**

- 1-5. (Canceled)
6. (Currently Amended) A formal equivalence verification method comprising:  
computing a Timed Binary Expression Diagram (TBED) for each circuit node as  
a Binary Expression Diagram (BED); and  
applying a satisfiability (SAT) solver or a BDD tool to establish equivalence for  
each TBED corresponding to each circuit ~~output~~ output node.
7. (Currently Amended) The formal equivalence verification method of claim 6,  
wherein ~~the~~ TBEDs of outputs of a specification circuit and an implementation circuit are  
compared.
8. (Original) The formal equivalence verification method of claim 7, wherein the  
circuit is a pipelined loop-free circuit.
9. (Currently Amended) A formal equivalence verification method comprising:  
listing ~~the~~ latches in a predetermined order;

representing ~~the~~ latch functions in a binary format; and  
computing a Timed Binary Expression Diagram (TBED) using ~~a~~the binary  
format.

10. (Original) The method of claim 9, wherein the binary format is a Binary Expression Diagram (BED) format.

11. (Original) The method of claim 9, wherein the TBED is converted into a format accepted by a Satisfiability (SAT) solver.

12. (Currently Amended) The method of claim 11, wherein if the SAT solver shows that ~~the~~ TBEDs of a plurality of circuits are equivalent then the circuits are also equivalent-as well.

13. (Original) The method of claim 12, wherein the equivalence determined by the SAT solver is a steady-state equivalence.

14. (Original) The method of claim 13, wherein the SAT solver determines combinational equivalence of Boolean formulas.

15. (Original) The method of claim 14, wherein the Boolean formulas comprise the TBEDs corresponding to specification and implementation outputs.

16. (Currently Amended) The method of claim 9, wherein ~~circuit~~ latches are traversed in a predetermined order and the latches are traversed to build TBEDs.

17. (Original) The method of claim 16, wherein the predetermined order selected and traversed is a bottom-up order that moves from inputs towards outputs.

18. (Original) The method of claim 17, wherein a relevant latch list is traversed once, according to the predetermined order.

19. (Currently Amended) The method of claim 16, wherein the TBED accounts for combinational properties on circuit inputs as well as any internal circuit nodes imposed by ~~the~~ a user.

20. (Currently Amended) A system for performing formal equivalence verification comprising a processor that defines a state transition for one or more latches in a circuit, places the latches in a predetermined order, represents the latches in a binary format, and computes a Timed Binary Expression Diagram (TBED) using ~~a~~ the binary format.

21. (Original) The system of claim 20 wherein the TBED is converted into a format accepted by a Satisfiability (SAT) solver.

22. (Currently Amended) A computer program product comprising:  
code that lists ~~the~~ latches in a predetermined order;  
code that represents ~~the~~ latch functions in a binary format; and  
code that computes a Timed Binary Expression Diagram (TBED) using ~~a~~ the  
binary format.

23. (Original) The computer program product of claim 22, wherein the binary format is a Binary Expression Diagram (BED) format.

24. (Original) The computer program product of claim 22, wherein the TBED is converted into a format accepted by a Satisfiability (SAT) solver.

25. (New) The computer program product of claim 23, wherein the BED format is a graphical representation of a Boolean formula.

26. (New) The formal equivalence verification method of claim 6, wherein the BED is a graphical representation of a Boolean formula.

Serial No. **10/608,423**

Docket No. **INTEL-0022**

Reply to Office Action dated October 6, 2005

27. (New) The method of claim 10, wherein the BED format is a graphical representation of a Boolean formula.